Machine Learning: V

- NN to process sequential data
  - Recurrent NN
  - LSTM & GRU
- Generative NN models
  - Autoencoders
  - GANs
- Take-home messages

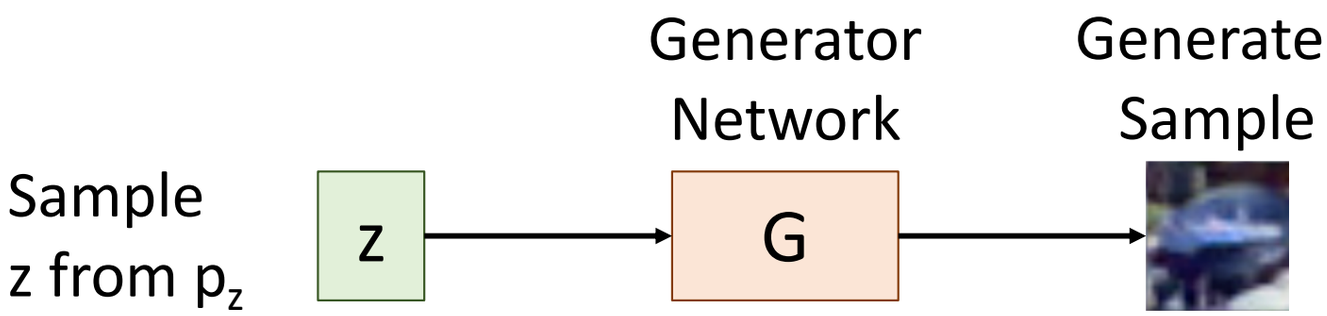
Generative Adversarial Networks

- Target: obtain a model for $p(x)$, then we can sample data from it.

Generative Adversarial Networks

- Target: obtain a model for $p(x)$, then we can sample data from it.

Idea: Introduce a latent variable z with simple prior p(z).
Sample $z \sim p(z)$ and pass to a **Generator Network** x = G(z)
Then x is a sample from the **Generator distribution** p_G. Want $p_G = p_{data}$!

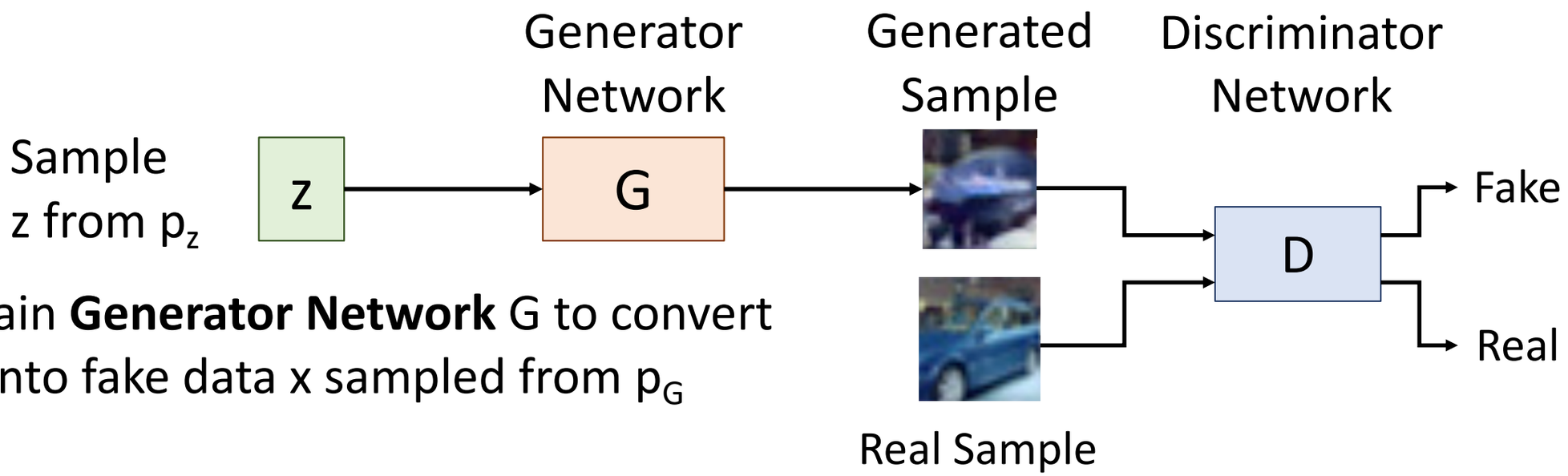


Train **Generator Network** G to convert
z into fake data x sampled from p_G

Generative Adversarial Networks

- Target: obtain a model for $p(x)$, then we can sample data from it.

Idea: Introduce a latent variable z with simple prior p(z).
Sample $z \sim p(z)$ and pass to a **Generator Network** x = G(z)
Then x is a sample from the **Generator distribution** p_G. Want $p_G = p_{data}$!

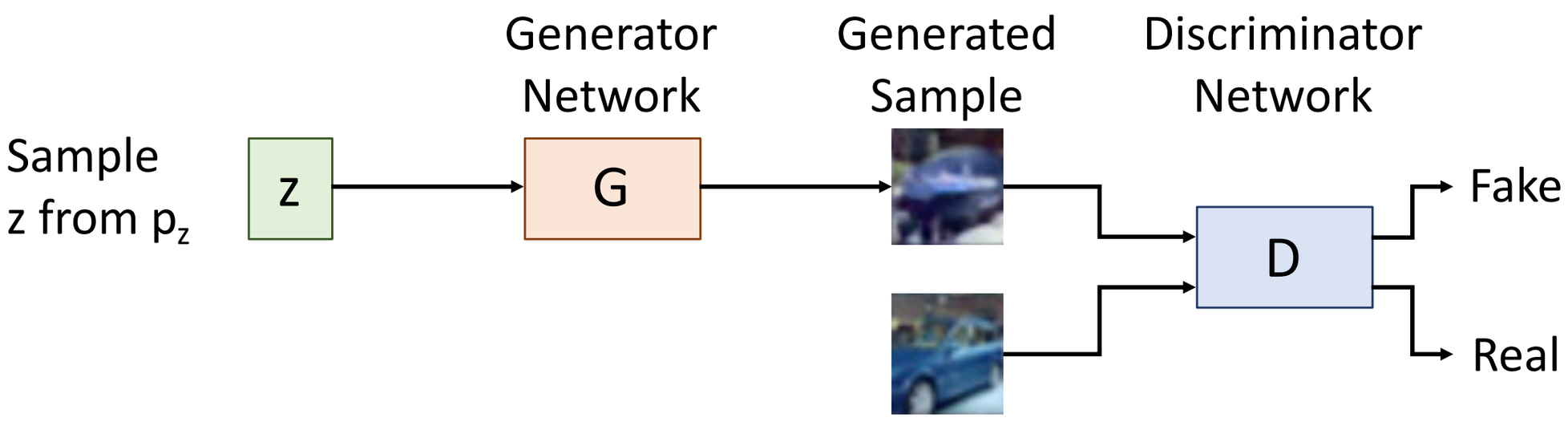


Train **Generator Network** G to convert z into fake data x sampled from p_G

The key idea is to train a discriminator to classify fake and real data.
A good generator should fool the discriminator to make its accuracy low:

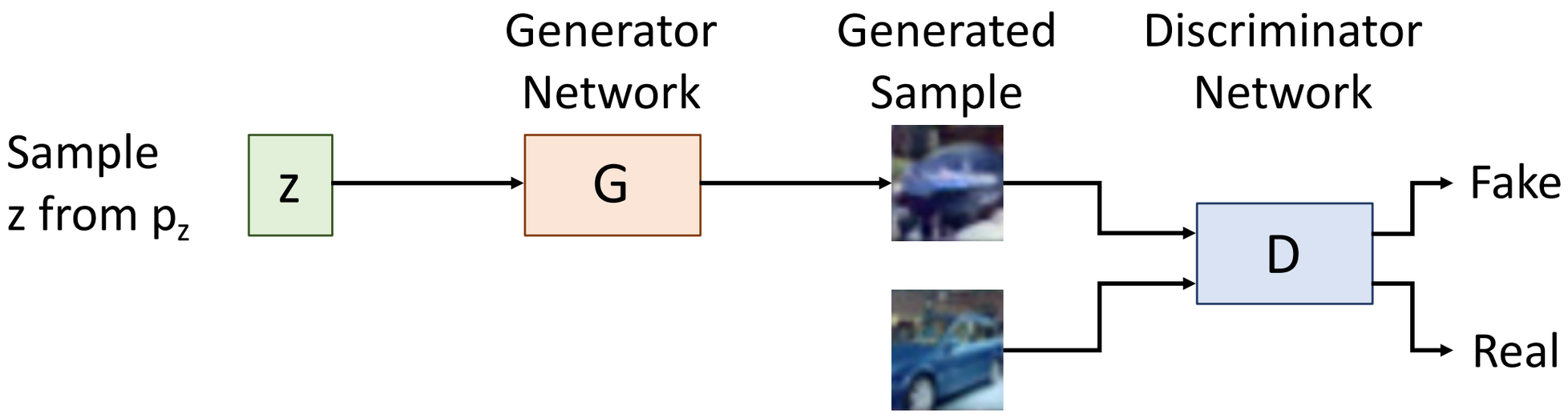
$$p_G = p_{data}$$

Adversarial Training



$$\min_{G} \max_{D} \left(E_{x \sim p_{data}} [\log D(x)] + E_{z \sim p(z)} \left[\log \left(1 - D(G(z)) \right) \right] \right)$$

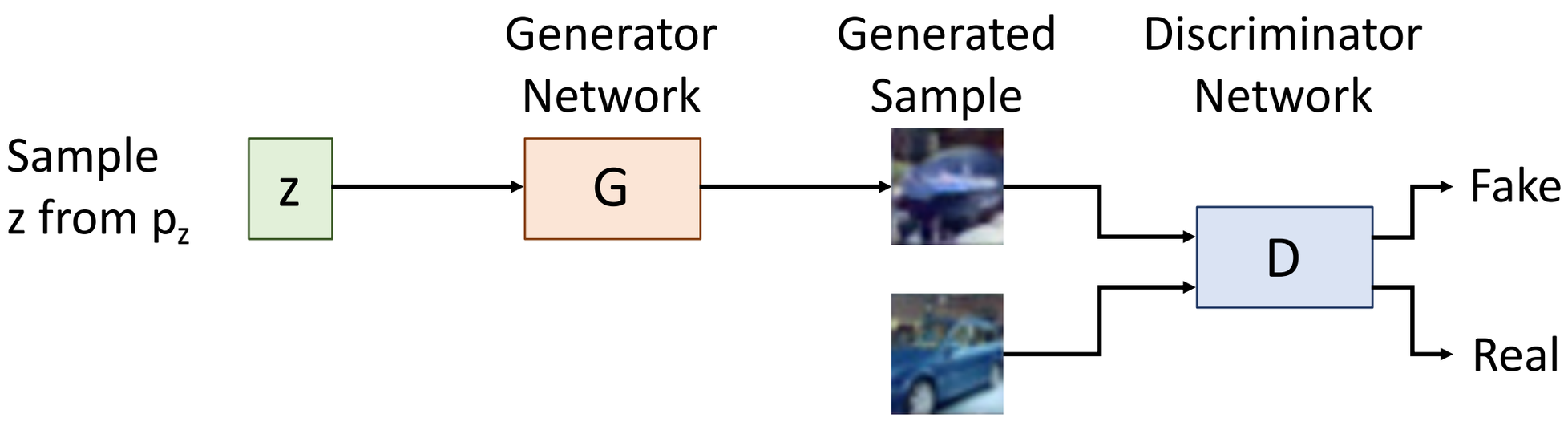
Adversarial Training



Discriminator wants D(x) = 1 for real data

$$\min_{G} \max_{D} \left(E_{x \sim p_{data}} [\log D(x)] + E_{z \sim p(z)} \left[\log \left(1 - D(G(z)) \right) \right] \right)$$

Adversarial Training



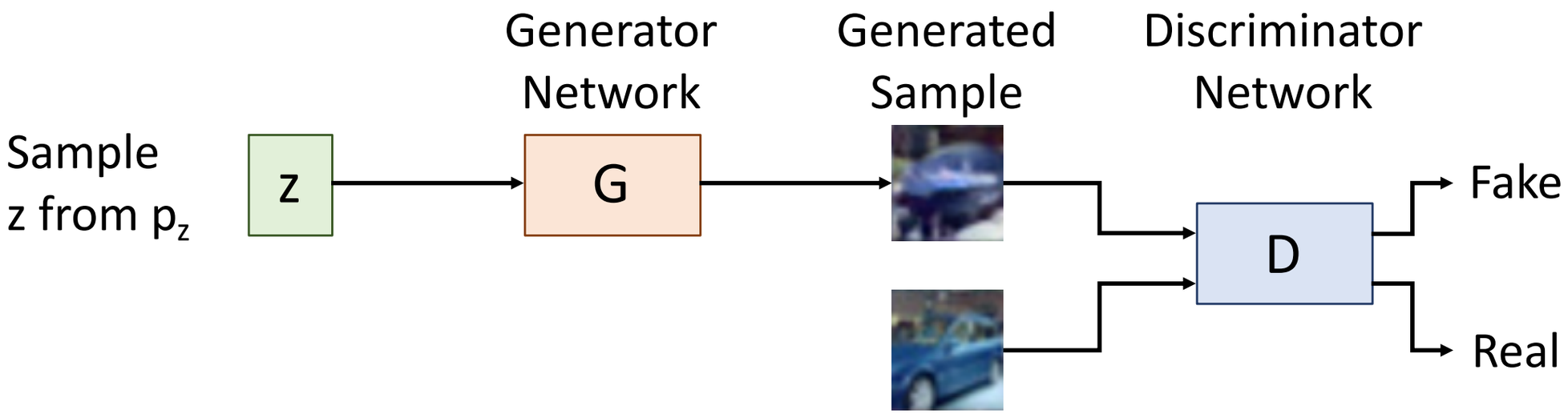
Discriminator wants D(x) = 1 for real data

Discriminator wants D(x) = 0 for fake data

$$\min_{G}\max_{D}\left(E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p(z)}\left[\log\left(1 - D(G(z))\right)\right]\right)$$

Generator wants D(x) = 1 for fake data

Adversarial Training



Discriminator wants D(x) = 1 for real data

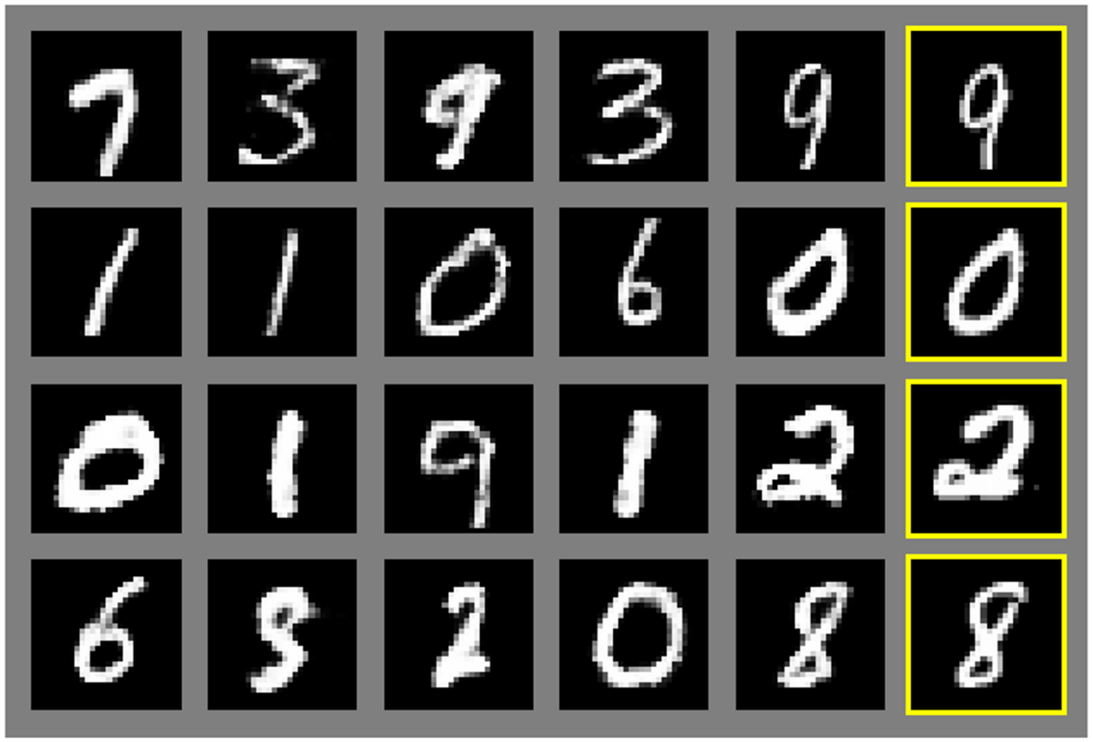
Discriminator wants D(x) = 0 for fake data

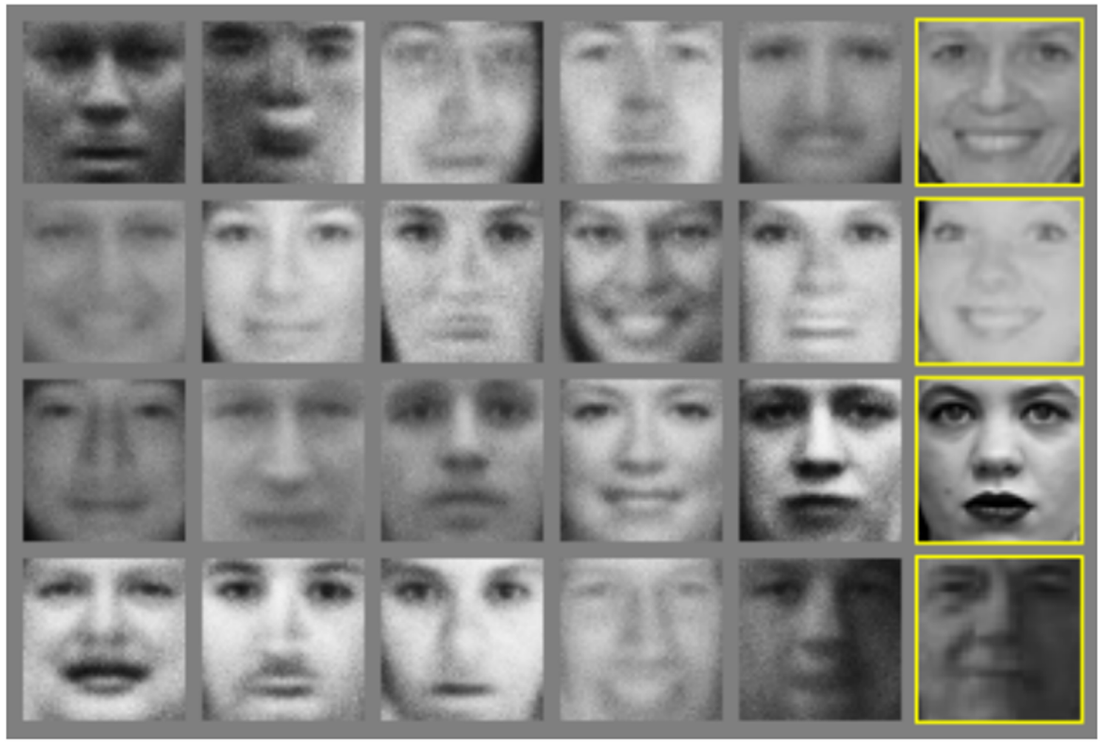
$$\min_{G}\max_{D}\left(E_{x\sim p_{data}}[\log D(x)] + E_{z\sim p(z)}\left[\log\left(1 - D(G(z))\right)\right]\right)$$

Generator wants D(x) = 1 for fake data

MinMax Game between generator and discriminator

Generation Results



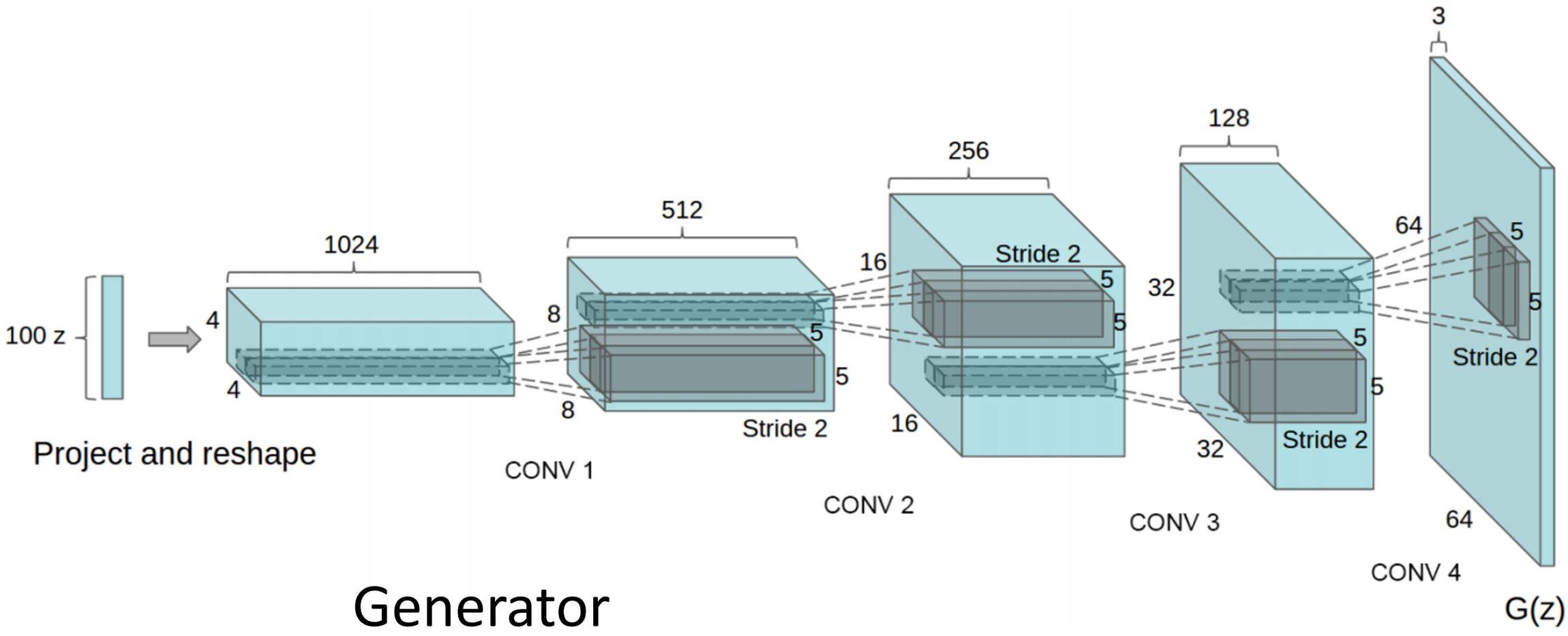


Nearest neighbor from training set

Goodfellow et al, “Generative Adversarial Nets”, NeurIPS 2014

Generation Results

Generative Adversarial Networks: DC-GAN



Radford et al, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, ICLR 2016

Generation Results

Generative Adversarial Networks: DC-GAN

Samples from the model look much better!

Radford et al, ICLR 2016

GAN's Latent Space

Generative Adversarial Networks: Interpolation

Interpolating between points in latent z space

Radford et al, ICLR 2016

GAN's Latent Space

Generative Adversarial Networks: Vector Math

Smiling woman

Neutral woman

Neutral man

Samples from the model

Radford et al, ICLR 2016

GAN's Latent Space

Generative Adversarial Networks: Vector Math

Smiling woman

Neutral woman

Neutral man

Samples from the model

Average Z vectors, do arithmetic

Radford et al, ICLR 2016

GAN's Latent Space

Generative Adversarial Networks: Vector Math

Smiling woman

Neutral woman

Neutral man

Samples from the model

Average Z vectors, do arithmetic

Smiling Man

Radford et al, ICLR 2016

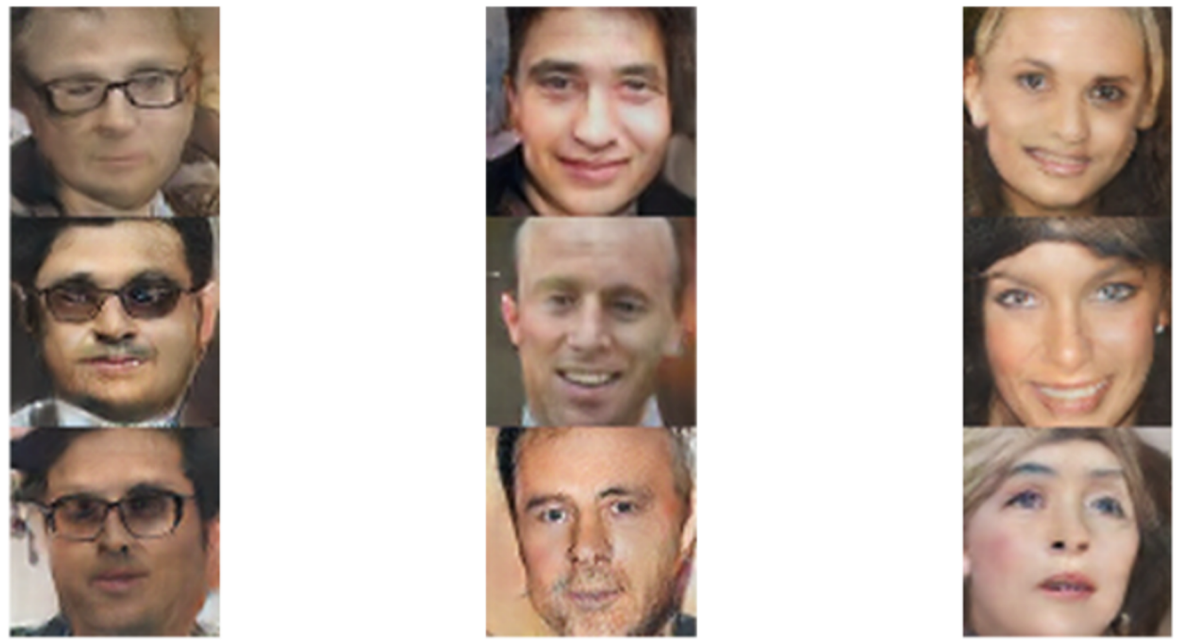
GAN's Latent Space

Man with glasses

Man w/o glasses

Woman w/o glasses

Samples from the model



Average Z vectors, do arithmetic

Radford et al, ICLR 2016

GAN's Latent Space

Man with glasses

Man w/o glasses

Woman w/o glasses

Samples from the model

Average Z vectors, do arithmetic

Woman with glasses

Radford et al, ICLR 2016

High-Resolution Generation

256 x 256 bedrooms

1024 x 1024 faces

Karras et al, “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, ICLR 2018

High-Resolution Generation

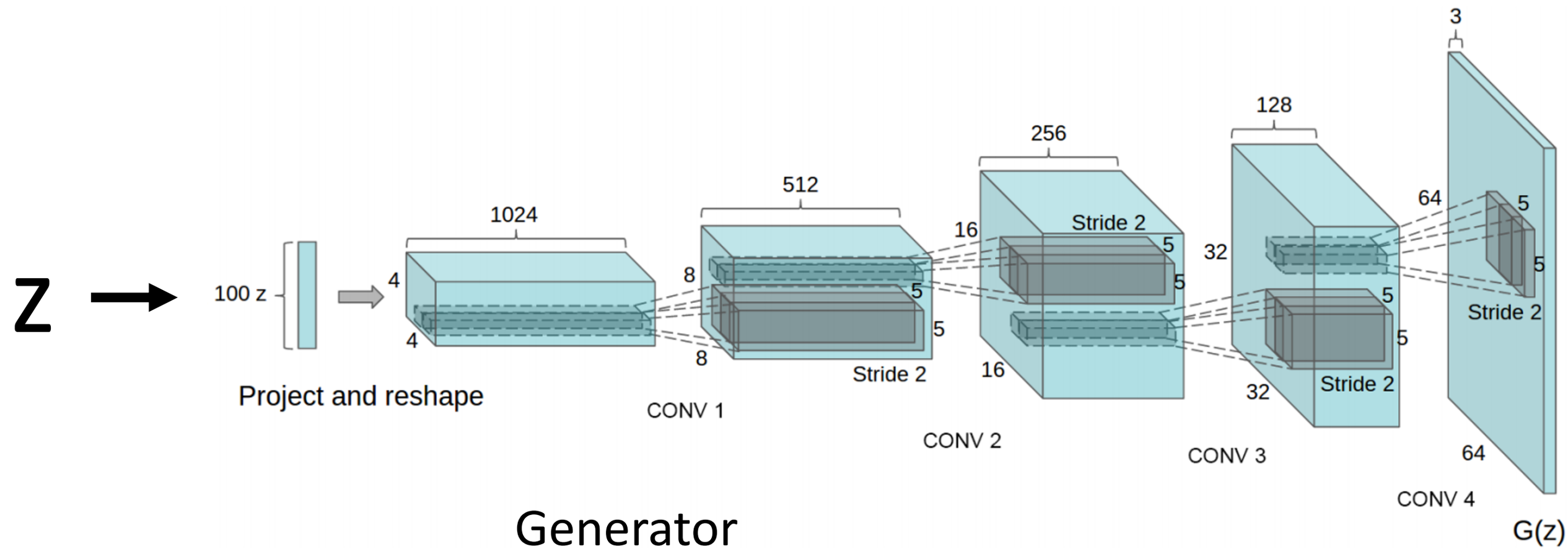
512 x 384 cars

1024 x 1024 faces

Karras et al, “A Style-Based Generator Architecture for Generative Adversarial Networks”, CVPR 2019

Conditional GANs

We can also make GAN to generate data under given context y

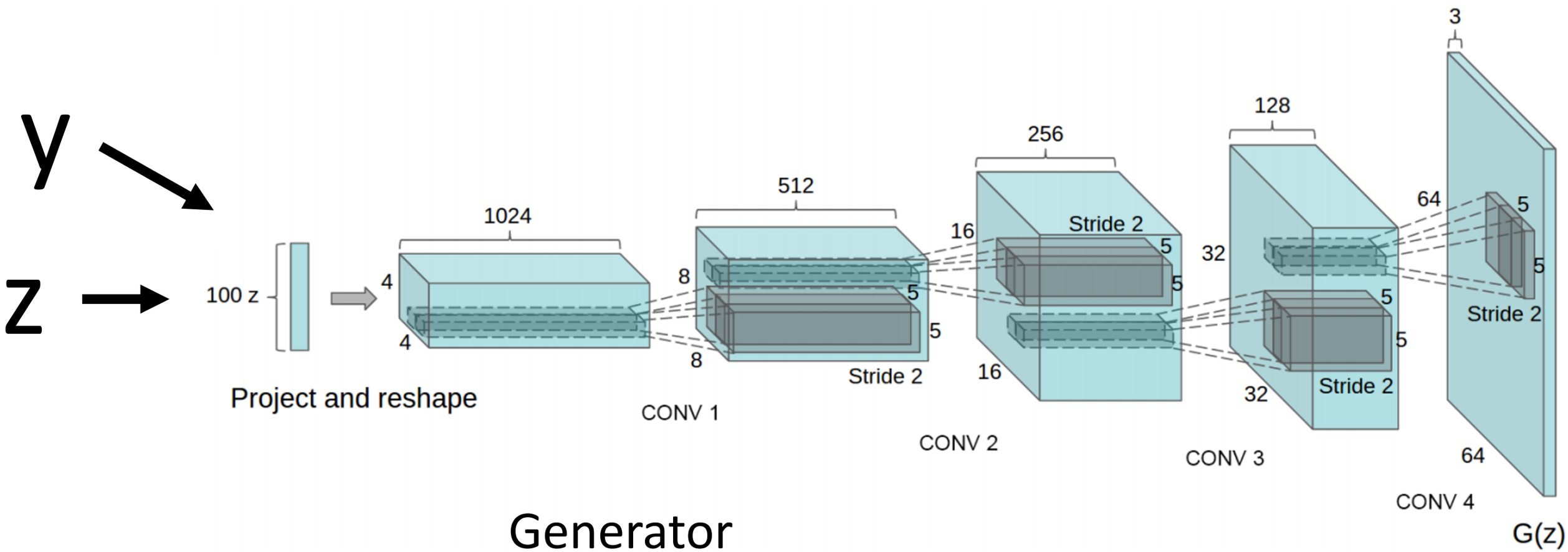


Generator

Radford et al, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, ICLR 2016

Conditional GANs

We can also make GAN to generate data under given context y



Generator

Radford et al, “Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks”, ICLR 2016

Text-to-Image Generation

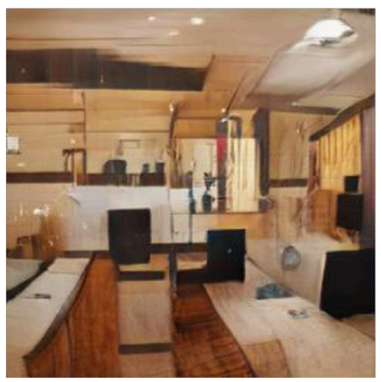
This bird is red and brown in color, with a stubby beak

The bird is short and stubby with yellow on its body

A bird with a medium orange bill white body gray wings and webbed feet

This small black bird has a short, slightly curved bill and long legs

A picture of a very clean living room



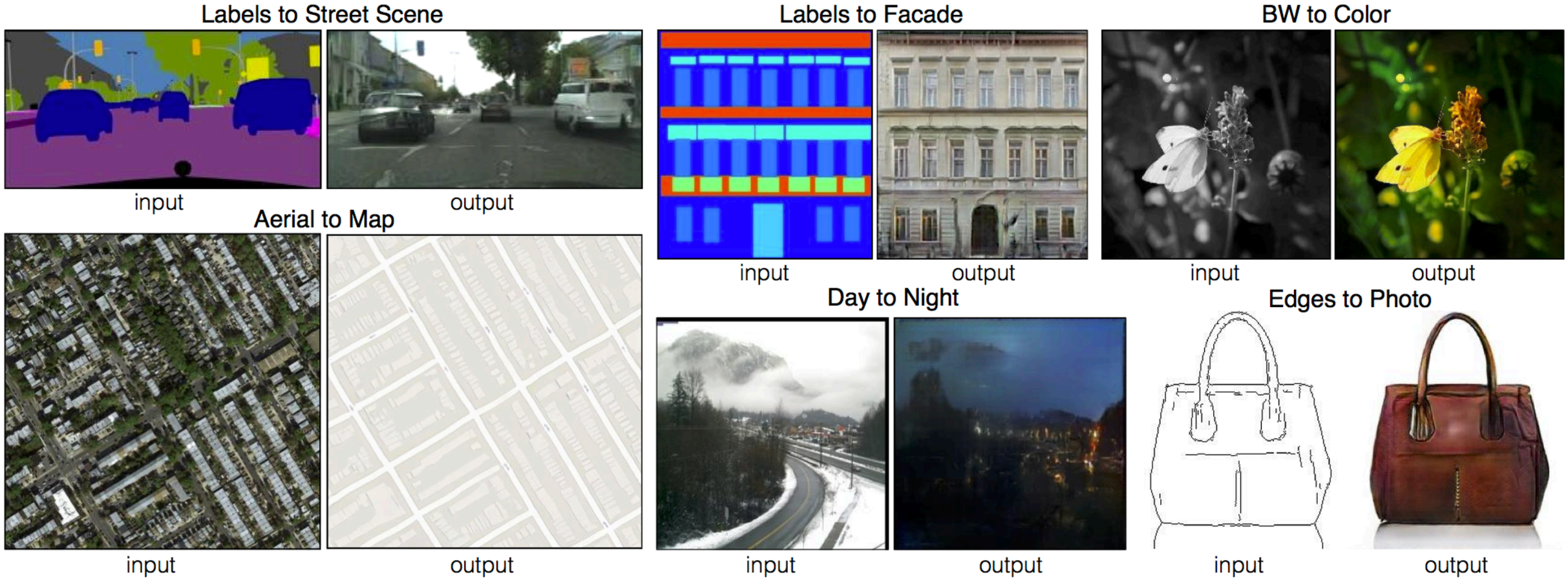
A group of people on skis stand in the snow

Eggs fruit candy nuts and meat served on white dish

A street sign on a stoplight pole in the middle of a day

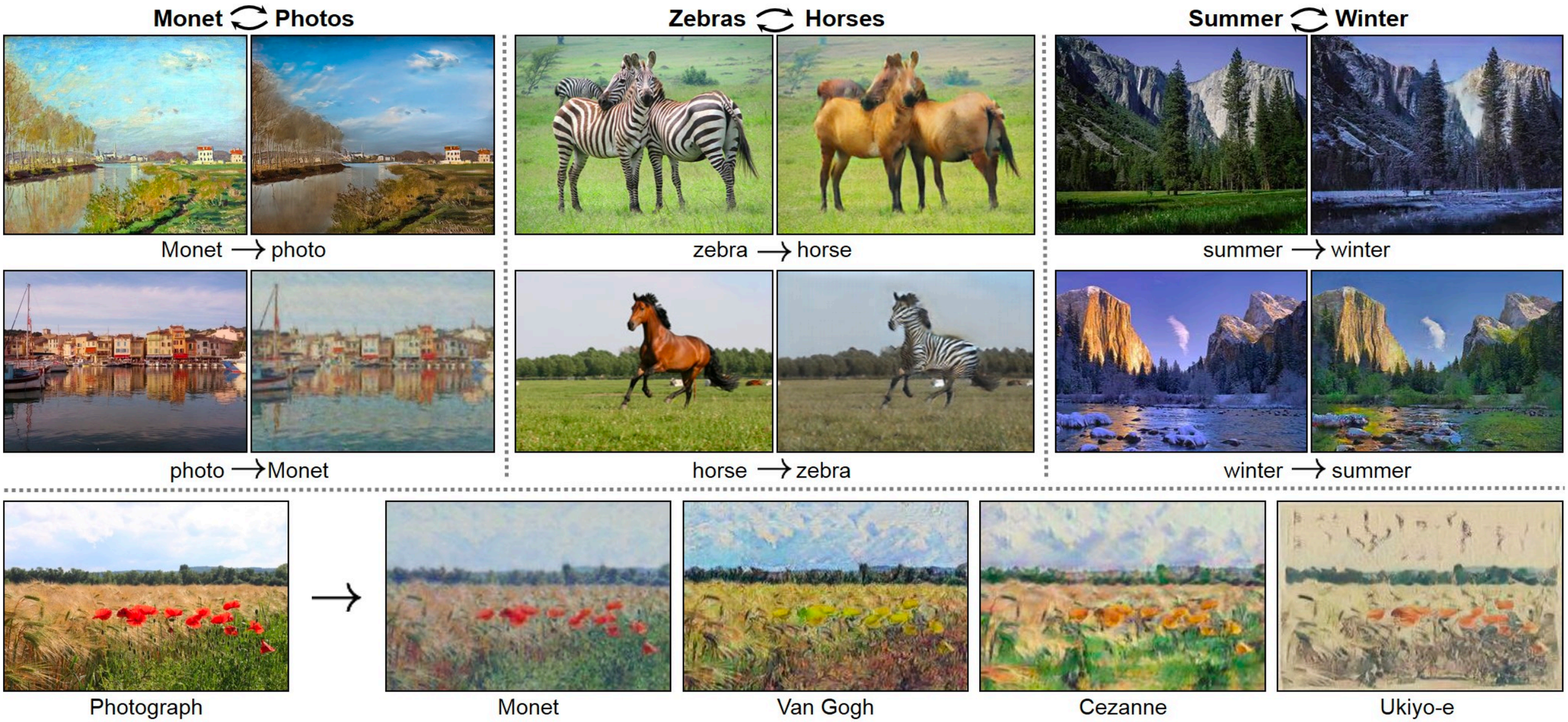
Zhang et al, “StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks.”, TPAMI 2018
Zhang et al, “StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks.”, ICCV 2017
Reed et al, “Generative Adversarial Text-to-Image Synthesis”, ICML 2016

Image-to-Image Translation



Isola et al, “Image-to-Image Translation with Conditional Adversarial Nets”, CVPR 2017

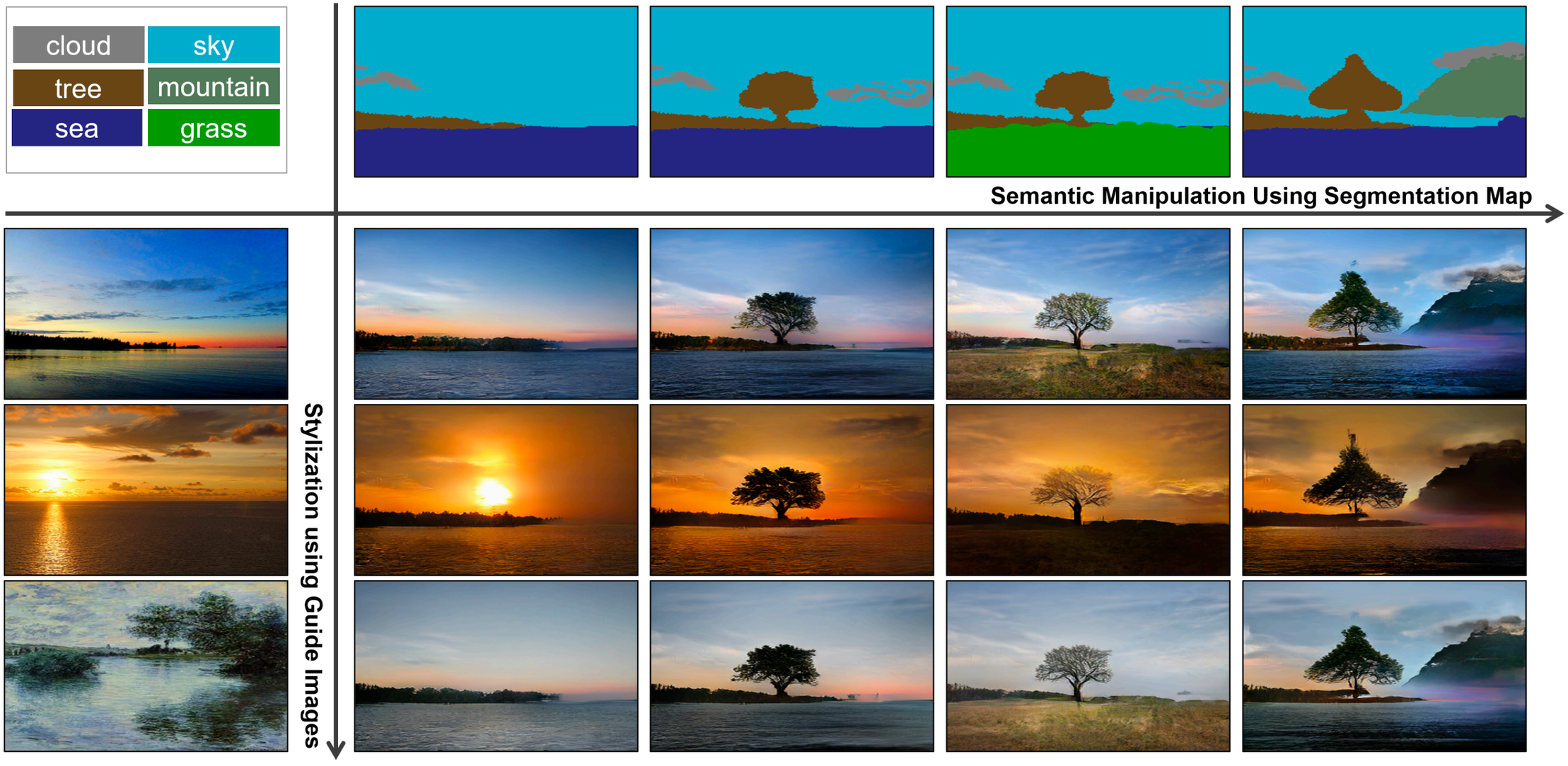
Image-to-Image Translation



Zhu et al, “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”, ICCV 2017

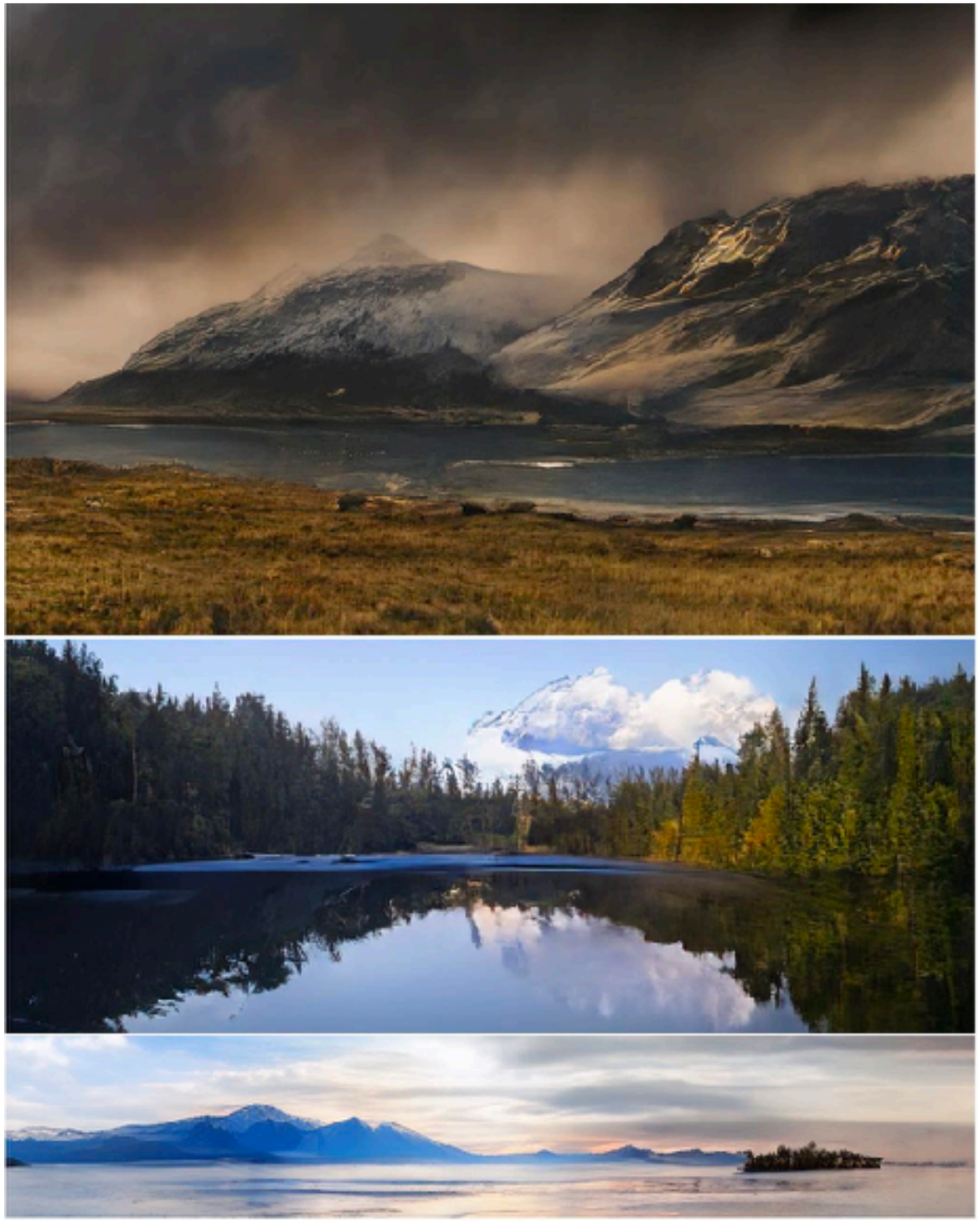
Other-to-Image Translation

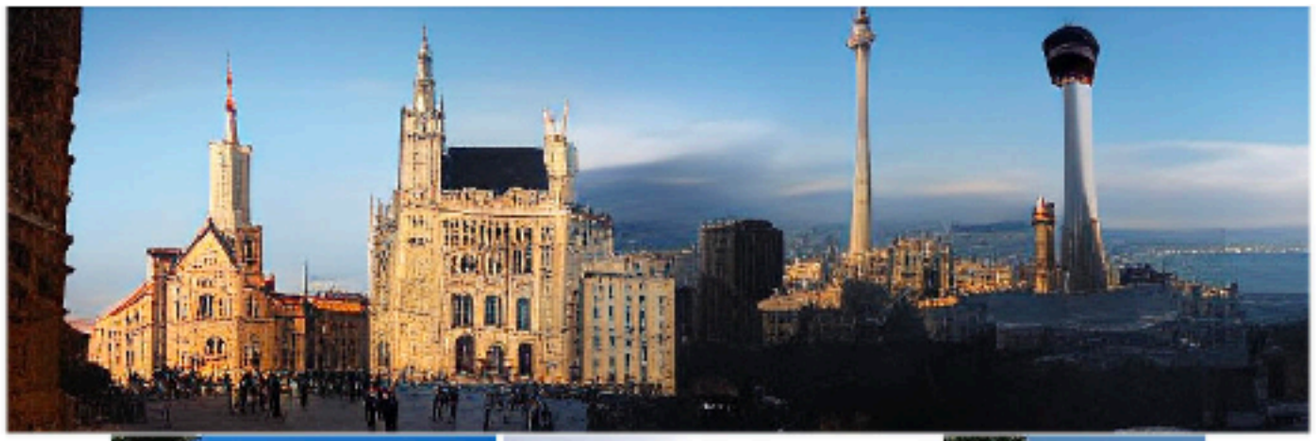
Label Map to Image

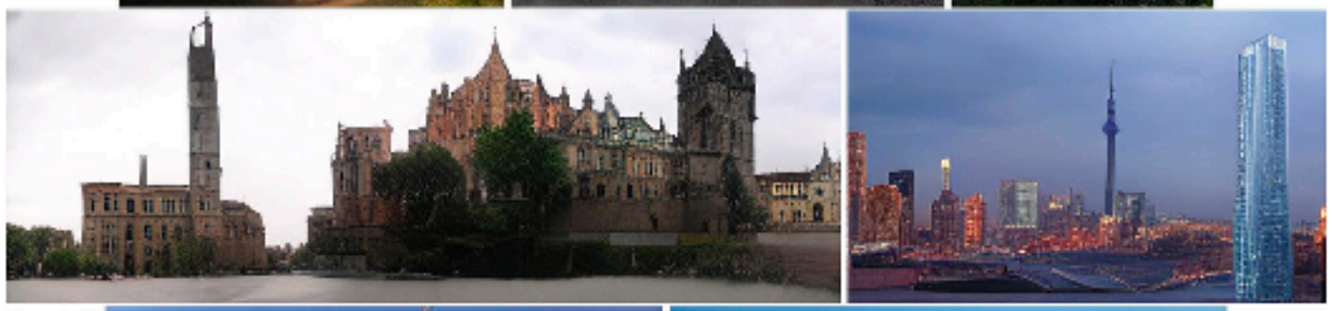


Park et al, “Semantic Image Synthesis with Spatially-Adaptive Normalization”, CVPR 2019

VQ-GAN

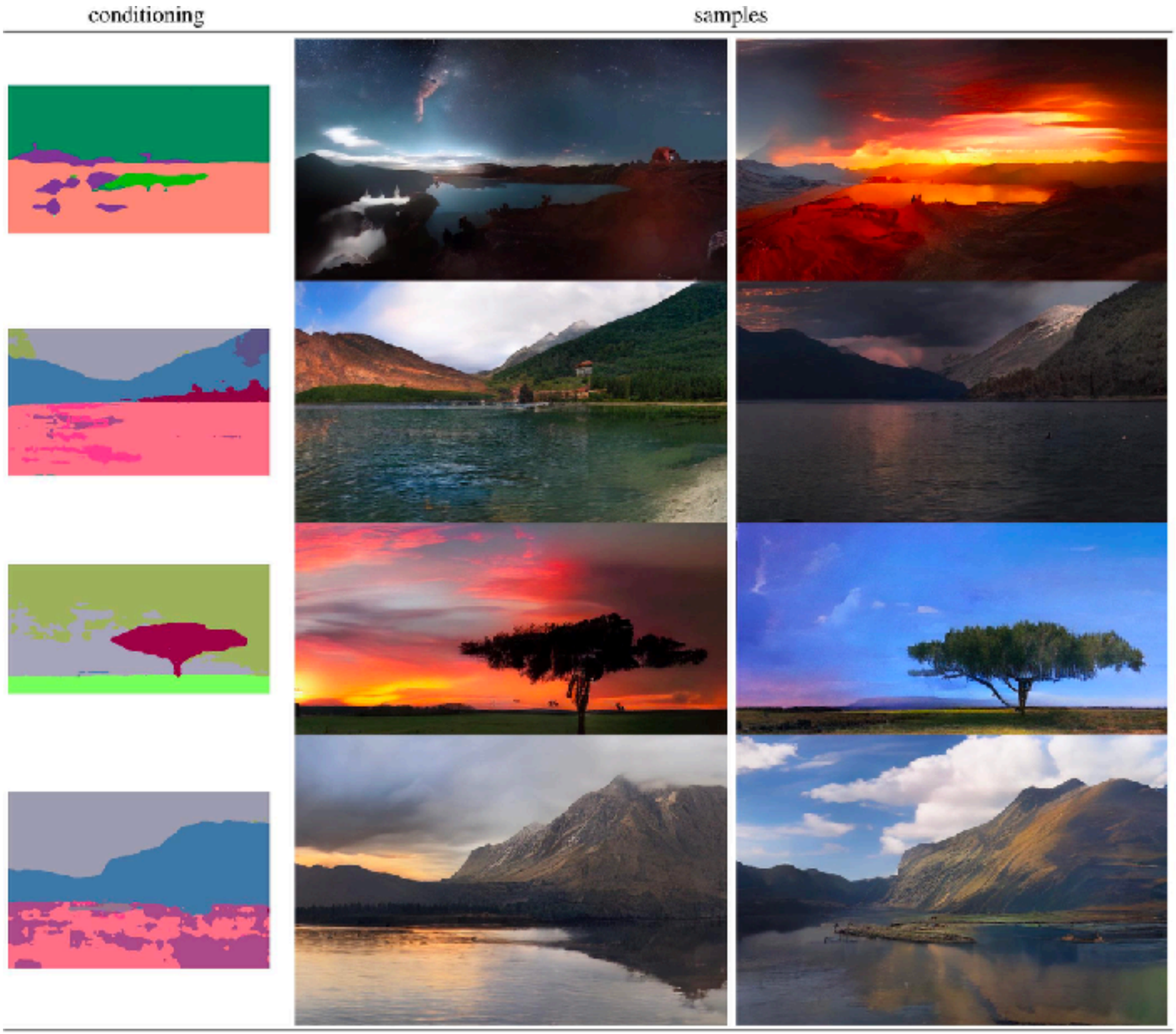






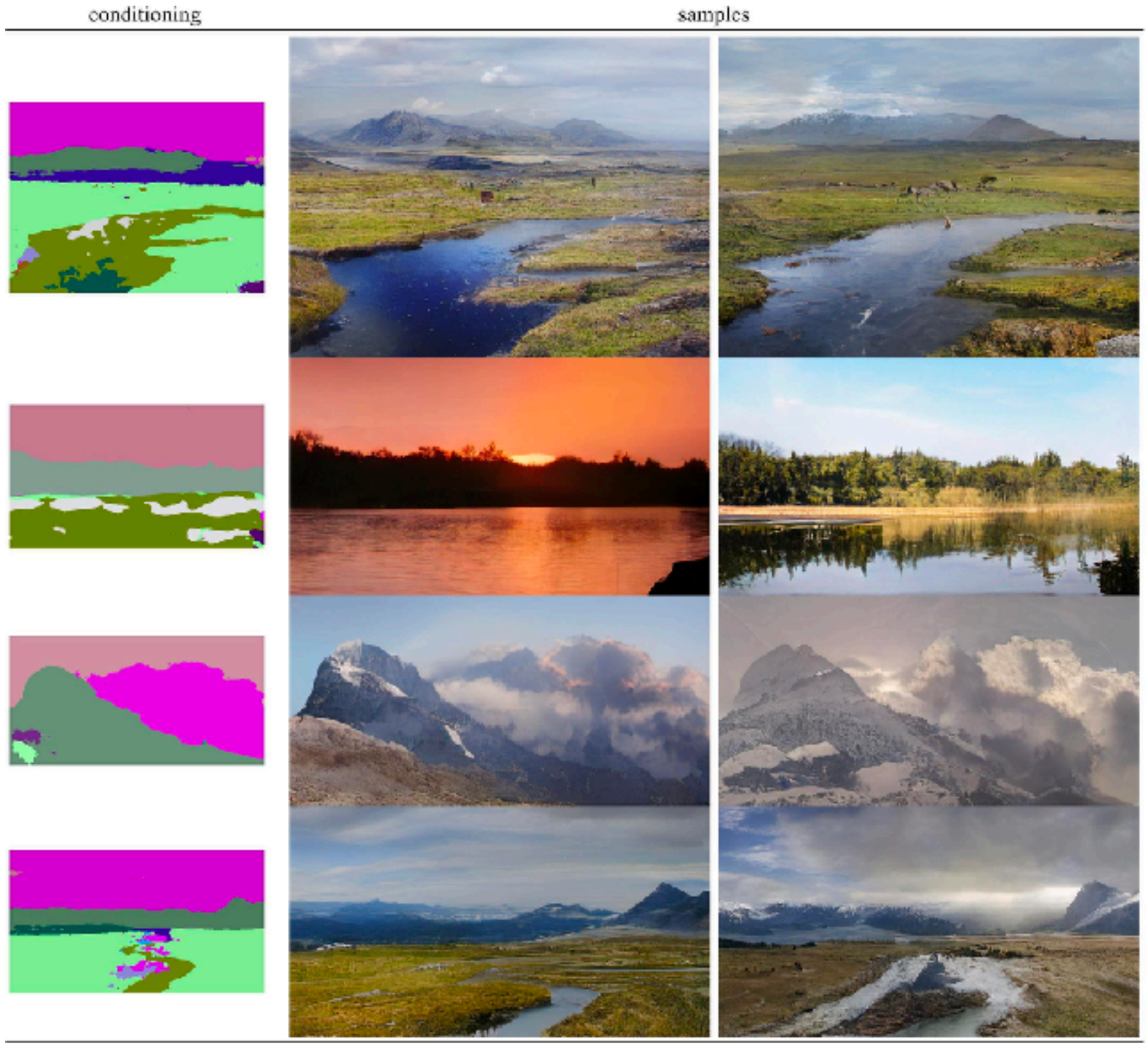
Taming Transformers for High-Resolution Image Synthesis, CVPR'21.

VQ-GAN



Taming Transformers for High-Resolution Image Synthesis, CVPR'21.

VQ-GAN



Taming Transformers for High-Resolution Image Synthesis, CVPR'21.

Text-to-Image with Diffusion Model

Sprouts in the shape of text 'Imagen' coming out of a fairytale book.

A photo of a Shiba Inu dog with a backpack riding a bike. It is wearing sunglasses and a beach hat.

A high contrast portrait of a very happy fuzzy panda dressed as a chef in a high end kitchen making dough. There is a painting of flowers on the wall behind him.

Teddy bears swimming at the Olympics 400m Butterfly event.

A cute corgi lives in a house made out of sushi.

A cute sloth holding a small treasure chest. A bright golden glow is coming from the chest.

Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding, NeurIPS'22.

Machine Learning: V

- NN to process sequential data
  - Recurrent NN
  - LSTM & GRU
- Generative NN models
  - Autoencoders
  - GANs
- Take-home messages

Take-Home Messages

- Recurrent neural networks (RNNs) are NNs whose outputs are also linked to the inputs. RNNs have hidden states storing information from a sequence of data.

- The major drawback of RNNs are not stable and not computational efficient. Modern models for sequential data: Transformers etc…

- Autoencoders compress information using an encoder and recover the information with a decoder. Their major advantage is to learn good representation of data from learning to compress and decompress information.

- GANs are based on the idea of adversarial training between generator and discriminator, leading to good generation quality.

Thanks for your attention!
Discussions?

Acknowledgement: Many materials in this lecture are taken from Justin Johnson: https://web.eecs.umich.edu/~justincj/teaching/eecs498/FA2020/schedule.html